

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11)特許出願公開番号

特開平10-293712

(43)公開日 平成10年(1998)11月4日

(51) Int.Cl.⁶
G 0 6 F 12/00

識別記号
531
535

F I
G O 6 F 12/00

531R
535Z

審査請求 有 請求項の数 5 FD (全 15 頁)

(21)出願番号 特願平9-116112

(22) 出願日 平成9年(1997)4月18日

(71) 出願人 000004237

日本電気株式会社

東京都港区芝五丁目7番1号

(72)発明者 杉山 誠治

東京都港区芝五丁目7番1号 日本電気株式会社内

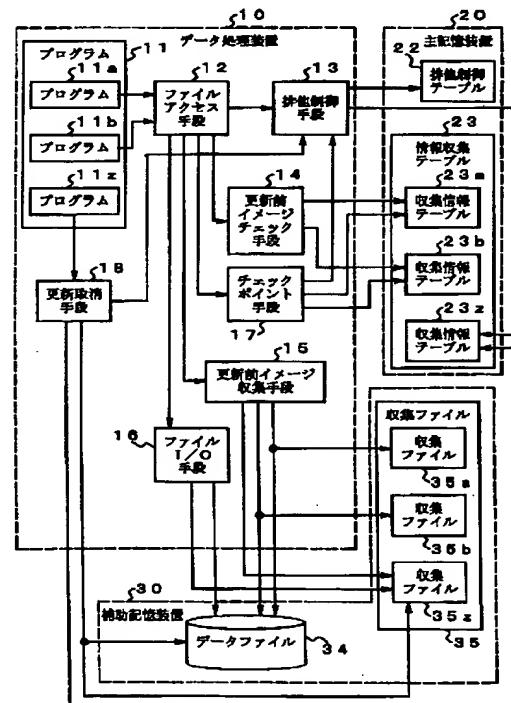
(74) 代理人 弁理士 松本 正夫

(54) 【発明の名称】 ファイルアクセス処理システム

(57) 【要約】 (修正有)

【課題】 ロールバック中のファイル更新の遅延を解消し、排他制御におけるスループット向上を実現する。

【解決手段】 データファイル34のロックとアンロックを行う排他制御手段13、ロック情報登録する排他制御テーブル22、ロック更新前イメージ収集手段15、更新前イメージを格納する収集ファイル35と、ロックの情報を格納する収集情報テーブル23、更新時に対象ロックの更新前イメージを収集すべきかをチェックする更新前イメージチェック手段14、データファイルに対するファイル読み書き手段16、更新完了指示により、更新前イメージを破棄するチェックポイント手段17と、更新取消要求により、収集ファイルのロックをデータファイルに戻す更新取消処理を行なう更新取消手段18を備える。



BEST AVAILABLE COPY

【特許請求の範囲】

【請求項1】 データファイルを分割して番号付けしたブロックを単位とした排他制御により、複数のプログラムから同時に同一のデータファイルのブロックに対して参照または更新のためのロック要求またはアンロック要求がなされた場合に、アクセス対象のブロックを前記ブロック番号によって管理し、前記ブロックのロックとアンロックを行う排他制御手段と、

前記排他制御手段によってロックされた前記ブロックの参照または更新ロックの種別を示すロック情報を登録する排他制御テーブルと、
対象ブロックの更新前に、前記ブロックの更新前イメージを収集する更新前イメージ収集手段と、
収集された前記更新前イメージを格納する収集ファイルと、

更新により収集された更新前イメージに対応する前記ブロックの情報を格納する収集情報テーブルと、
対象ブロックの更新時に、前記収集情報テーブルを参照し、対象ブロックの更新前イメージを収集すべきかをチェックする更新前イメージチェック手段と、

前記データファイルに対する対象ブロックの読み込みあるいは書き出しを行うファイル読み書き手段と、
更新完了指示により、ロックしていた対象ブロックをアンロックし更新前イメージを破棄するチェックポイント手段と、

更新取消要求により、更新ロックされたブロックのアンロックを前記排他制御手段に要求し、前記収集情報テーブルに基づいて前記収集ファイルのブロックを前記データファイルに戻す更新取消処理を行なう更新取消手段とを備えることを特徴とするファイルアクセス処理システム。

【請求項2】 前記排他制御手段は、参照または更新のためのロック要求がなされた場合であって、ロック要求された前記ブロックのロック情報が他のプログラム用の前記排他制御テーブルに登録されている場合、

前記ロック要求と前記ロック情報が参照どうしであれば、前記ロック要求されたブロックのロック情報を前記排他制御テーブルに追加し、

参照どうしでなければ、前記ロック要求されたブロックが更新取消処理中であるかをチェックし、更新取消処理中でなければ、前記ブロックのアンロックを待ち合わせ、更新取消処理中であれば、前記収集情報テーブルから前記ブロックの前記収集ファイル上の位置情報を取得し、前記ロック要求が更新であれば、前記収集情報テーブルの前記ブロックの情報を削除して前記ロック要求されたブロックのロック情報を前記排他制御テーブルに追加し、前記ロック要求が参照であれば、前記ロック要求されたブロックのロック情報を前記排他制御テーブルに追加することを特徴とする請求項1に記載のファイルアクセス処理システム。

【請求項3】 前記排他制御テーブルに、複数の前記プログラム毎に、前記ブロックの参照または更新ロックのロック情報と、更新取消処理中の有無を示す情報を登録する複数のテーブルを備え、

前記排他制御手段は、前記ロック要求されたブロックが更新取消処理中であるかをどうかを、前記プログラム毎のテーブルの更新取消処理中の有無を示す情報によって判別することを特徴とする請求項2に記載のファイルアクセス処理システム。

10 【請求項4】 前記排他制御テーブルに、複数の前記プログラム毎に、前記ブロックの参照または更新ロックのロック情報と、更新取消処理中の有無を示す情報を登録する複数のテーブルを備え、

前記排他制御手段は、アンロック要求がなされた場合、更新取消処理前のアンロックかどうかをチェックし、更新取消処理前のアンロックであれば、前記排他制御テーブルのアンロック要求を出したプログラムのテーブルに更新取消処理中を示す情報を追加し、参照ロックしていたブロックについてのみ前記テーブルの参照ロック情報を削除し、

更新取消処理前のアンロックでなければ、前記排他制御テーブルのアンロック要求を出したプログラムのテーブル上の全ブロックのロック情報及び更新取消処理中を示す情報を削除することを特徴とする請求項1に記載のファイルアクセス処理システム。

【請求項5】 前記排他制御手段は、参照または更新のためのロック要求がなされた場合であって、ロック要求された前記ブロックのロック情報が他のプログラム用の前記排他制御テーブルに登録されている場合、

30 前記ロック要求と前記ロック情報が参照どうしであれば、前記ロック要求されたブロックのロック情報を前記排他制御テーブルに追加し、

参照どうしでなければ、前記ロック要求されたブロックが更新取消処理中であるかをチェックし、更新取消処理中でなければ、前記ブロックのアンロックを待ち合わせ、更新取消処理中であれば、前記収集情報テーブルから前記ブロックの前記収集ファイル上の位置情報を取得し、前記ロック要求が更新であれば、前記収集情報テーブルの前記ブロックの情報を削除して前記ロック要求さ

40 れたブロックのロック情報を前記排他制御テーブルに追加し、前記ロック要求が参照であれば、前記ロック要求されたブロックのロック情報を前記排他制御テーブルに追加し、

前記更新前イメージチェック手段は、前記収集情報テーブルに前記ブロックの情報を格納されていなければ、前記収集情報テーブルに前記ブロックの情報を追加し、

前記更新前イメージ収集手段は、前記収集情報テーブルに追加された前記ブロックの更新前イメージを、前記他のプログラム用の前記収集ファイルから読み出し、前記

50 ロック要求を出したプログラム用の前記収集ファイルに

書き出し、

前記ファイルI/O手段は、更新であれば、前記ブロックを前記データファイルに書き出し、参照であれば、前記ロック要求を出したプログラム用の前記収集ファイルまたは前記データファイルから前記ブロックを読み込むことを特徴とする請求項1に記載のファイルアクセス処理システム。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、コンピュータにおけるファイルアクセス処理システムに関し、特に同一のデータファイルを多数のプログラムが矛盾なく参照、更新する処理に関して有効な排他制御を行なうファイルアクセス処理システムに関する。

【0002】

【従来の技術】以下、具体的に従来の排他制御方式について説明する。

【0003】従来、ファイルの排他制御は、1つのデータファイルを複数のプログラムが同時に更新、参照する場合、ファイル内容に矛盾が生じないようにデータファイルを一定のブロックに分割し、番号付けたブロックを単位とした排他制御を行っていた。

【0004】すなわち、更新したブロックは更新ロックという状態となり、更新ロックされたブロックはプログラムが意識して更新完了指示を出さない限り他のプログラムからは更新、参照できず待ち状態となる。また、参照したブロックは参照ロックという状態となり、参照ロックされたブロックは同様に更新完了指示を出さない限り他のプログラムから参照できるが更新することはできなかった。

【0005】ここで、更新完了指示とは、プログラムが一連の参照、更新を完了してファイル内容が論理的に正しい時にOSに出すべき指示で、これによりロックされていたブロックは全てアンロックされ、ロックされていたブロックに対する他のプログラムからの参照、更新が可能となる。以下、この更新完了指示のことをチェックポイントと記述する。

【0006】上記のことからファイルの排他制御によってデータファイルに対するアクセスに遅延が発生することがあり、スループット低下の一原因となっていた。この遅延を軽減するため、様々な提案がなされている。

【0007】例えば、特開平3-123946号公報の「データベースの排他制御方法」では、データファイルにアクセスするプログラム(=トランザクション)を、参照及び更新を行う更新プログラムと参照のみを行う検索プログラムとに分類し、更新プログラムは二相ロックを用いた多版同時実行制御によって管理し、検索プログラムは基本タイムスタンプ順序を用いた多版同時実行制御によって管理する構成となっている。

【0008】二相ロックを用いた多版同時実行制御と

は、あるブロックに対して更新した時は更新前と更新後の二つの内容を持ち、もし更新ロック中のブロックに対して参照が行われた場合は、更新前のイメージを参照させ、その更新ロックのアンロック時に参照ロックを保証ロックと呼ばれる参照も更新も不可能なロックに変更する。

【0009】また、基本タイムスタンプ順序を用いた多版同時実行制御とは、あるブロックに対する参照時にはリードスタンプと呼ばれる時間情報をブロックの情報と

10 格納し、更新時にはライトスタンプと呼ばれる時間情報をブロックの情報として格納し、またプログラム自身も開始した時間情報をタイムスタンプを持ち、これらを用いることにより矛盾なく排他制御を行う。

【0010】つまり、ブロックに対する更新毎に新しいブロックの内容が累積され、あるブロックに対する参照、更新はアクセスするプログラムのタイムスタンプより前で最大のライトスタンプを持つものが選択され、それが最新のデータのときに更新を許可し、そのブロックのリードスタンプがプログラムのタイムスタンプより前20 なら更新を許可し、ブロックに対しては同時に更新ロックできないという規則に従う更新時の参照を目的とした方式である。このようにしてできる限り遅延を少なくすることを目的としていた。

【0011】また、プログラム異常時のデータファイルの不完全な更新をなくすため、データファイル更新時にプログラム毎、データファイル毎に用意された収集ファイルに対象ブロックの更新前の内容(以下、更新前イメージと記述)をコピー(以下、収集すると記述)し、プログラム異常時やシステム異常時等に自動でデータファイルに書き戻す更新取消(以下ロールバックと記述)の機能がある。

【0012】ファイルの排他制御と同時にロールバックの機能が働くとき、ロールバックもデータファイルの更新の一環であるため、それまで更新ロックしていたブロックはロールバック処理が完了するまでアンロックされなかった。

【0013】また、チェックポイントを取ると前述のロックしているブロックのアンロックとともに、収集ファイル中の更新前イメージの破棄も行うこととしていた。40 このロールバックの機能は、排他制御においてデッドロック(プログラムがブロックを取り合って永久遅延に陥ること)発生時にも用いられる。

【0014】

【発明が解決しようとする課題】上述した従来の排他制御方式では、以下に述べるような種々の問題点を有していた。

【0015】第1の問題点は、あるプログラムのロールバック中はロールバック中のブロックに更新アクセスできず、ロールバックに時間がかかるとスループットが低下することがあった。その理由は、従来の技術がロール

バックのことを考慮せず、通常の更新と同様に扱っていたためである。

【0016】第2の問題点は、多数の更新プログラムが同一のファイルにアクセスしている時、更新プログラムが中止させられる可能性が高くなり、かえって合計処理時間が長くなることがあった。その理由は、更新プログラムどうしは二相ロックを用いた多版同時実行制御を行い、更新ロック中のブロックに更新プログラムが参照を行った場合に更新完了処理時に保証ロックされ、参照したブロックを更新しようとすると古い版のブロックを参照しているため、プログラムを中止せざるをえないからである。

【0017】第3の問題点は、多数のプログラムが同一のブロックに対してアクセスすると、多数のブロックのイメージが作成され、データベースのサイズが増大する場合があった。その理由は、古い版を参照しているプログラムが存在する限り更新ロックが解除されても参照しているイメージが残るためである。

【0018】本発明の目的は、上記従来の問題点を解決し、ロールバック中のファイル更新の遅延を解消し、デッドロック発生時や、プログラム異常終了時の排他制御におけるスループット向上を実現するファイルアクセス処理システムを提供することにある。

【0019】本発明の他の目的は、更新プログラムを中止しなければならない状態に陥ることをなくし、処理時間の増加を回避することの可能なファイルアクセス処理システムを提供することにある。

【0020】本発明のさらに他の目的は、ロールバックに用いる収集ファイルによるデータファイルのサイズの増大を防止できるファイルアクセス処理システムを提供することにある。

【0021】

【課題を解決するための手段】上記目的を達成する本発明のファイルアクセス処理システムは、データファイルを分割して番号付けしたブロックを単位とした排他制御により、複数のプログラムから同時に同一のデータファイルのブロックに対して参照または更新のためのロック要求またはアンロック要求がなされた場合に、アクセス対象のブロックを前記ブロック番号によって管理し、前記ブロックのロックとアンロックを行う排他制御手段と、前記排他制御手段によってロックされた前記ブロックの参照または更新ロックの種別を示すロック情報を登録する排他制御テーブルと、対象ブロックの更新前に、前記ブロックの更新前イメージを収集する更新前イメージ収集手段と、収集された前記更新前イメージを格納する収集ファイルと、更新により収集された更新前イメージに対応する前記ブロックの情報を格納する収集情報テーブルと、対象ブロックの更新時に、前記収集情報テーブルを参照し、対象ブロックの更新前イメージを収集すべきかをチェックする更新前イメージチェック手段と、

前記データファイルに対する対象ブロックの読み込みあるいは書き出しを行うファイル読み書き手段と、更新完了指示により、ロックしていた対象ブロックをアンロックし更新前イメージを破棄するチェックポイント手段と、更新取消要求により、更新ロックされたブロックのアンロックを前記排他制御手段に要求し、前記収集情報テーブルに基づいて前記収集ファイルのブロックを前記データファイルに戻す更新取消処理を行なう更新取消手段とを備える。

10 【0022】請求項2の本発明のファイルアクセス処理システムは、前記排他制御手段は、参照または更新のためのロック要求がなされた場合であって、ロック要求された前記ブロックのロック情報が他のプログラム用の前記排他制御テーブルに登録されている場合、前記ロック要求と前記ロック情報が参照どうしであれば、前記ロック要求されたブロックのロック情報を前記排他制御テーブルに追加し、参照どうしでなければ、前記ロック要求されたブロックが更新取消処理中であるかをチェックし、更新取消処理中でなければ、前記ブロックのアンロックを待ち合わせ、更新取消処理中であれば、前記収集情報テーブルから前記ブロックの前記収集ファイル上の位置情報を取得し、前記ロック要求が更新であれば、前記収集情報テーブルの前記ブロックの情報を削除して前記ロック要求されたブロックのロック情報を前記排他制御テーブルに追加し、前記ロック要求が参照であれば、前記ロック要求されたブロックのロック情報を前記排他制御テーブルに追加することを特徴とする。

【0023】請求項3の本発明のファイルアクセス処理システムは、前記排他制御テーブルに、複数の前記プログラム毎に、前記ブロックの参照または更新ロックのロック情報と、更新取消処理中の有無を示す情報を登録する複数のテーブルを備え、前記排他制御手段は、前記ロック要求されたブロックが更新取消処理中であるかをどうかを、前記プログラム毎のテーブルの更新取消処理中の有無を示す情報によって判別することを特徴とする。

【0024】請求項4の本発明のファイルアクセス処理システムは、前記排他制御テーブルに、複数の前記プログラム毎に、前記ブロックの参照または更新ロックのロック情報と、更新取消処理中の有無を示す情報を登録する複数のテーブルを備え、前記排他制御手段は、アンロック要求がなされた場合、更新取消処理前のアンロックかどうかをチェックし、更新取消処理前のアンロックであれば、前記排他制御テーブルのアンロック要求を出したプログラムのテーブルに更新取消処理中を示す情報を追加し、参照ロックしていたブロックについてのみ前記テーブルの参照ロック情報を削除し、更新取消処理前のアンロックでなければ、前記排他制御テーブルのアンロック要求を出したプログラムのテーブル上の全ブロックのロック情報及び更新取消処理中を示す情報を削除することを特徴とする。

【0025】請求項5の本発明のファイルアクセス処理システムは、前記排他制御手段は、参照または更新のためのロック要求がなされた場合であって、ロック要求された前記ロックのロック情報を他のプログラム用の前記排他制御テーブルに登録されている場合、前記ロック要求と前記ロック情報を参照どうしであれば、前記ロック要求されたロックのロック情報を前記排他制御テーブルに追加し、参照どうしでなければ、前記ロック要求されたロックが更新取消処理中であるかをチェックし、更新取消処理中でなければ、前記ロックのアンロックを待ち合わせ、更新取消処理中であれば、前記収集情報テーブルから前記ロックの前記収集ファイル上の位置情報を取得し、前記ロック要求が更新であれば、前記収集情報テーブルの前記ロックの情報を削除して前記ロック要求されたロックのロック情報を前記排他制御テーブルに追加し、前記ロック要求が参照であれば、前記ロック要求されたロックのロック情報を前記排他制御テーブルに追加し、前記更新前イメージチェック手段は、前記収集情報テーブルに前記ロックの情報を格納されていなければ、前記収集情報テーブルに前記ロックの情報を追加し、前記更新前イメージ収集手段は、前記収集情報テーブルに追加された前記ロックの更新前イメージを、前記他のプログラム用の前記収集ファイルから読み出し、前記ロック要求を出したプログラム用の前記収集ファイルに書き出し、ファイル読み書き手段は、更新であれば、前記ロックを前記データファイルに書き出し、参照であれば、前記ロック要求を出したプログラム用の前記収集ファイルまたは前記データファイルから前記ロックを読み込むことを特徴とする。

【0026】

【発明の実施の形態】次に、本発明の実施の形態について図面を参照して詳細に説明する。図1を参照すると、本発明の実施の形態は、プログラム制御により動作するデータ処理装置10と、半導体で構成される主記憶装置20と、ディスク装置などで構成される補助記憶装置30とを備える。

【0027】データ処理装置10は、プログラム11と、ファイルアクセス手段12と、排他制御手段13と、更新前イメージチェック手段14と、更新前イメージ収集手段15と、ファイルI/O手段16と、チェックポイント手段17と、更新取消手段18を備えて構成される。

【0028】主記憶装置20は、排他制御テーブル22と、収集情報テーブル23とを備える。また、補助記憶装置30は、一定の大きさのブロックに分割管理されているデータファイル34と、収集ファイル35とを備える。

【0029】排他制御テーブル22は、ロックしているブロックの情報を格納するためのテーブルであり、ロックしているプログラム毎に、ロック番号、ロック種別

(参照ロックか更新ロックかを示す)、ロールバック中を示す表示等の情報を格納している。

【0030】収集情報テーブル23は、プログラム11の更新により、更新前イメージ収集手段15が収集した更新前イメージの情報を格納するためのテーブルであり、収集ファイルの位置、収集したロック番号、それに対応する収集ファイルのロック番号等の情報を格納している。

【0031】収集ファイル35は、プログラム11の更新により、更新前イメージ収集手段15が収集した更新前イメージを格納するためのファイルである。

【0032】プログラム11がデータファイル34に参照、更新のためにアクセスする時、ファイルアクセス手段12は最初に排他制御手段13を動作させる。

【0033】排他制御手段13は、アクセス対象となるブロックの番号を排他制御テーブル22を基にチェックし、そのブロックがロック可能かどうかの判断をする。ロック不可能な場合は対象ブロックがアンロックされるまで待ち状態となり、ロック可能な場合、排他制御テーブル22に対象ブロックのロック番号、ロック種別等の情報を登録してロックする。

【0034】ファイルアクセス手段12は、アクセスの種類(読み込みか書き出しか)に応じて、読み込みの時はファイルI/O手段16を動作させ、書き出しの時は更新前イメージチェック手段14を動作させる。

【0035】更新前イメージチェック手段14は、要求元プログラムに対応する収集情報テーブル23から対象ブロックの更新前イメージがすでに収集されているかどうかの判断をする。

【0036】ファイルアクセス手段12は、参照の場合や、対象ブロックの更新前イメージがすでに収集されている場合は、ファイルI/O手段16を動作させ、まだ収集されていない場合は更新前イメージ収集手段15を動作させる。

【0037】更新前イメージ収集手段15は、対象ブロックの内容をデータファイル34から読み込むと共に、プログラム11に対応する収集ファイル35にその内容を書き出す。

【0038】ファイルI/O手段16は、要求に従ってデータファイル34から対象ブロックの内容を読み込んだり、データファイル34の対象ブロックにデータを書き込んだりする。

【0039】また、プログラム11がチェックポイントを取った場合、ファイルアクセス手段12はチェックポイント手段17を動作させる。

【0040】チェックポイント手段17は、排他制御手段13にアンロック要求を出し、収集ファイル35にある全ての更新前イメージを破棄し、収集情報テーブル23を初期化する。

【0041】また、プログラム11から更新取消要求が

出された場合、更新取消手段18が排他制御手段13を動作させ、ロールバック開始前のアンロック要求を出し、収集情報テーブル23に基づいて収集ファイル35の内容をデータファイル34へロールバックを行い、ロールバック完了後に排他制御手段13を動作させ、改めてアンロック要求を出す。

【0042】本発明の実施の形態における特徴は、更新取消手段18がロックしてあったブロックを、従来技術ではロールバック終了までアンロックしなかったのに対し、ロールバック開始前にアンロックする要求を排他制御手段13に出すことにより、ロールバック対象ブロックにも他プログラムからアクセスを可能とすることである。

【0043】以下に、図1及び図2～8を参照して上記のように構成される本発明の実施の形態の動作について詳細に説明する。

【0044】プログラム11がデータファイル34をアクセスしたり、チェックポイントを取ろうとする要求は、ファイルアクセス手段12が受け取る。ファイルアクセス手段12の処理を示す図2において、ファイルアクセス手段12はこの要求を判定し（ステップ201）、参照、更新などのファイルアクセス要求ならば排他制御手段13を動作させ（ステップ202）、アクセス対象ブロックのロック要求を出し、チェックポイント要求ならばチェックポイント手段17を動作させる（ステップ208）。

【0045】排他制御手段13の処理を示す図3において、起動された排他制御手段13は、ロックかアンロックか要求を判定し（ステップ301）、ロック要求ならば排他制御テーブル22の他プログラム用のテーブルに対象ブロックが存在するかチェックする（ステップ302）。排他制御テーブル22の他プログラム用のテーブルに対象ブロックが存在しない場合、すなわち対象ブロックが他プログラムによってロックされていなければアクセス可能となり、アクセス種別によって排他制御テーブル22に更新ロック、あるいは参照ロックとして対象ブロックを追加する（ステップ303、304、305）。

【0046】もし、排他制御テーブル22の他のプログラム用テーブルに対象ブロックが存在したとき、参照どうしのアクセスは可能であるため、アクセスを許可する（ステップ306）。更新どうし、あるいは参照と更新の場合は、対象ブロックを更新ロックしているプログラムにロールバック中の表示があるかどうかチェックし（ステップ307）、ロールバック中の表示がないものがある場合は同時アクセス不可能なため、他プログラムのブロックのアンロックを待つ（ステップ308）。

【0047】対象ブロックを更新ロックしているプログラム全てにロールバック中の表示がある場合は、同時アクセス可能となり、対象ブロックの収集ファイル35上

の位置を収集情報テーブル23から得る（ステップ309）。そして、アクセス種別をチェックして参照する場合は排他制御テーブル22に参照ロックとして追加し（ステップ310、304）、更新する場合はそのブロックに対してロールバックが行われないようするため、他プログラムの収集情報テーブル23上の対象ブロックの情報を削除し（ステップ311）、排他制御テーブル22に更新ロックとして追加する（ステップ305）。

10 【0048】また、ステップ301においてアンロック要求ならば、ロールバック開始前のアンロックかどうかをチェックし（ステップ312）、ロールバック開始前のアンロックならば排他制御テーブル22上の対象プログラムにロールバック中の表示を追加して参照ロックしていたブロックだけは排他制御テーブル22から削除してアンロックし（ステップ314）、そうでなければ排他制御テーブル上の対象プログラムの全ブロックの情報及びロールバック中の表示を削除し（ステップ313）、ロックしていたブロックをアンロックする。処理20が完了したらファイルアクセス手段12に戻る。

【0049】図2において、ファイルアクセス手段12は、排他制御手段13から戻ると、アクセスの種類に応じて参照ならファイルI/O手段16を動作させ（ステップ203、207）、更新なら更新前イメージチェック手段14を動作させる（ステップ204）。

【0050】更新前イメージチェック手段14の処理を示す図4において、更新前イメージチェック手段14は、収集情報テーブル23に更新対象ブロックの情報があるかどうかチェックし（ステップ401）、対象ブロックの情報がない場合は更新前イメージを収集する必要があるため、収集情報テーブル23に対象ブロックの情報を追加し（ステップ402）、対象ブロックの情報がある場合は何もせずにファイルアクセス手段12に戻る。

【0051】図2において、ファイルアクセス手段12は、更新前イメージチェック手段14から戻ると、更新前イメージを収集済みならばファイルI/O手段16を動作させ（ステップ205、207）、更新前イメージを収集する必要があるなら更新前イメージ収集手段15を動作させる（ステップ206）。

【0052】更新前イメージ収集手段15の処理を示す図5において、更新前イメージ収集手段15は、対象ブロックが他プログラム用の収集ファイル35に存在するかどうかチェックし（ステップ501）、収集ファイル35に存在する場合は、対象ブロックの更新前イメージをその収集ファイルから読み込み（ステップ502）、収集ファイル35に存在しない場合は対象ブロックをデータファイル35から読み込む（ステップ504）。それぞれ読み込んだ更新前イメージは収集ファイル35に書き出し（ステップ503）、ファイルアクセス手段1

2に戻る。

【0053】図2において、ファイルアクセス手段12は、更新前イメージ収集手段15から戻ったらファイルI/O手段16を動作させる。

【0054】ファイルI/O手段16の処理を示す図6において、ファイルI/O手段16は、まず参照かどうかをチェックし(ステップ601)、参照の場合は、対象ブロックの存在するファイルに従って、対象ブロックが収集ファイル35にある場合は収集ファイル35から(ステップ602、603)、データファイル34にある場合はデータファイル34から読み込む(ステップ602、ステップ604)。また、更新する場合はデータファイル34の対象ブロックへ書き出す(ステップ605)。

【0055】図2において、ファイルアクセス手段12は、プログラム11からチェックポイント要求を受けると、チェックポイント手段17を動作させる(ステップ208)。

【0056】チェックポイント手段17の処理を示す図7において、チェックポイント手段17は、まず排他制御手段13を動作させ(ステップ701)、ロックしていた全ブロックをアンロックする。次に、収集情報テーブル23を初期化して(ステップ702)、更新前イメージを全て破棄する。

【0057】次に、更新取消手段18の動作を図8に基づいて示す。プログラム11の異常終了時等に、自動あるいはプログラム11が意識してデータファイル14に対する更新取消要求を出した時、更新取消手段18が動作する。

【0058】更新取消手段18は、まず排他制御手段13を動作させ(ステップ801)、ロールバック開始前のアンロック要求を出し、参照、更新でロックしてあったブロックを他プログラムからアクセス可能とした後、ロールバックを行う。すなわち、収集情報テーブル23にあるブロックを一つずつ順に収集ファイル35からデータファイル34へ書き戻す(ステップ802)。そして、ロールバックが完了したら排他制御手段13を動作させ(ステップ803)、アンロック要求を出し、ロックしていたブロックを実際に排他制御テーブル22から削除させる。

【0059】上述したように、本発明の実施の形態によれば、ロールバック開始前にそれまでロックしていたブロックをアンロックし、ロールバック対象ブロックは収集ファイルに対してアクセスさせることにより、ロールバック対象ブロックへの同時更新が可能となる。

【0060】従来では、ロールバック中はロールバック対象ブロックにはロールバック完了まで参照、更新等のアクセスができず、特開平3-123946号公報の方法でも、参照プログラムは参照しかできず待ち状態となり、更新プログラムが参照、更新しようとすると更新ブ

ログラムが中止されることになっていたが、本実施の形態による方式では、そのような遅延がなくなり、ロールバック中の排他制御のスループットが向上するものである。

【0061】また、本実施の形態による方式は、ロールバック対象ブロックに対するアクセス時、参照はもちろん、更新もロールバック対象の収集ファイルに対しては参照しかせず、かつそのブロックのアクセス時の内容はロールバック完了後の内容と同一であるため、ファイル10内容の論理的正当性は保たれ、ファイル内容の矛盾は発生しない。さらに、ロールバックの機能を利用することにより、新たにファイルスペースをとる必要がなく、補助記憶装置の容量を圧迫しない。

【0062】

【実施例】以下本発明の一実施例について図を参照しながら詳細に説明する。

【0063】本実施例においては、図1のプログラム11に含まれるプログラム11a、11b、11zが図12に示す内容の同一のデータファイル34(ブロック番号21～35で構成される)をアクセスしており、図9はその時の排他制御テーブル22の内容を示している。

【0064】ここで、プログラム11aはデータファイル34のブロック番号23、25、26にそれぞれアクセス済みであり、プログラム11bはブロック番号27、29にそれぞれアクセス済みである。また、プログラム11zはブロック番号30、31、32にアクセスした後、プログラムが異常終了し、ロールバックを開始した状態であるとする。同様に、図10は収集情報テーブル23の内容を、図11は収集ファイル35の内容を30それぞれ示している。

【0065】図1及び図9～12の状態の時、プログラム11zは更新取消手段18を動作させ、さらに更新取消手段18は排他制御手段13を動作させ(ステップ801)、ロールバック開始前のアンロック要求を出し、それが完了した直後を表している。つまり、排他制御手段13がアンロック要求であり(ステップ301)、ロールバック開始前のアンロックのため(ステップ312)、排他制御テーブル22のプログラム11z用のテーブルにロールバック中の表示をつけて参照ロックして40いたブロックだけは排他制御テーブル22から削除してアンロックし(ステップ314)、更新取消手段18に戻った状態である。

【0066】同様に図示の状態の時、プログラム11aがデータファイル34のブロック番号31を参照しようとした時、ファイルアクセス手段12は排他制御手段13を動作させる(ステップ201、202)。排他制御手段13はロック要求であるため(ステップ301)、排他制御テーブル22にブロック番号31があるかを調べる(ステップ302)。このとき、排他制御テーブル22のプログラム11z用のテーブルにブロック番号350

1は存在するため、ロールバック中の表示を調べると（ステップ306、307）、ロールバック中の表示があり、アクセス可能となる。

【0067】そして、実際にアクセスすべき収集ファイル上の位置を収集情報テーブル23zから取得し（ステップ309）、最後に排他制御テーブル22のプログラム11a用のテーブルにブロック番号31を参照として追加し（ステップ304）、ファイルアクセス手段12に戻る。

【0068】ファイルアクセス手段12は、アクセス種別が参照のため（ステップ203）、ファイルI/O手段16を動作させる（ステップ207）。ファイルI/O手段16はアクセスは参照であり（ステップ601）、ブロック番号31は収集ファイル35z上に存在するため（ステップ602）、収集ファイル35zからデータファイル34のブロック番号31に対応するブロック番号42zを読み込み（ステップ303）、プログラム11aに戻る。

【0069】同様に図示の状態の時、プログラム11bがデータファイル34のブロック番号22を参照しようとした時、ファイルアクセス手段12はまず排他制御手段13を動作させる（ステップ201、202）。排他制御手段13は、ロック要求であるため（ステップ301）、排他制御テーブル22にブロック番号22があるか調べる（ステップ302）。ブロック番号22は存在しないため、排他制御テーブル22のプログラム11b用のテーブルにブロック番号22を参照として追加し（ステップ303、304）、ファイルアクセス手段12に戻る。

【0070】ファイルアクセス手段12は、アクセスが参照であるため（ステップ203）、ファイルI/O手段16を動作させる（ステップ207）。ファイルI/O手段16はアクセス種別が参照であり（ステップ601）、ここで対象ブロックはデータファイル34に存在するため（ステップ602）、データファイル34からブロック番号22の内容を読み込み（ステップ604）、プログラム11bに戻る。

【0071】同様に図示の状態の時、プログラム11aがブロック番号32を更新しようとした時、ファイルアクセス手段12は排他制御手段13を動作させる（ステップ201、202）。排他制御手段13はロック要求であるため（ステップ301）、排他制御テーブル22にブロック番号32があるかチェックする（ステップ302）。ブロック番号32はプログラム11z用のテーブルにあり、さらにロールバック中の表示が付いているため（ステップ306、307）、アクセス可能と判断する。

【0072】ブロック番号32の収集ファイル35上の位置（収集ファイル35zのブロック番号43z）を収集情報テーブル23zから取得し（ステップ309）、

そのブロックはロールバックされないように収集情報テーブル23zからブロック番号32の情報を削除し（ステップ310、311）、最後に排他制御テーブル22のプログラム11a用のテーブルへブロック番号32を更新として追加し（ステップ305）、ファイルアクセス手段12へ戻る。

【0073】ファイルアクセス手段12は、更新であるため、更新前イメージチェック手段14を動作させる（ステップ203、204）。更新前イメージチェック

10 手段14は、収集情報テーブル23aの全ての情報を調べるが、ブロック番号32は登録されていないため（ステップ401）、収集情報テーブル23aにブロック番号32を追加し（ステップ402）、ファイルアクセス手段12に戻る。

【0074】ファイルアクセス手段12は、ブロック番号32が収集されていないため、更新前イメージ収集手段15を動作させる（ステップ205、206）。更新前イメージ収集手段15は、ブロック番号32を収集ファイル35zのブロック番号43zから読み込み、その20 内容を収集ファイル35aのブロック番号43aに書き出し（ステップ501、502、503）、ファイルアクセス手段12に戻る。ファイルアクセス手段12は、ファイルI/O手段16を動作させ（ステップ207）、データファイル34を更新する。

【0075】ファイルI/O手段16は、更新であるため、プログラム11aで指定された内容をデータファイル34のブロック番号32へ書き出し（ステップ601、605）、プログラム4aに戻る。

【0076】図示の状態の時、プログラム11bがブロ30 ック番号21に書き出し要求を出した時、ファイルアクセス手段12は排他制御手段13を動作させる（ステップ201、202）。排他制御手段13は、ロック要求であるため（ステップ301）、排他制御テーブル22を調べるが（ステップ302）、ブロック番号21は存在しないため、アクセス可能と判断する。更新であるため（ステップ303）、排他制御テーブル22のプログラム11b用のテーブルにブロック番号21を更新として追加し（ステップ305）、ファイルアクセス手段12に戻る。

【0077】ファイルアクセス手段12は、次に更新前イメージチェック手段14を動作させる（ステップ203、204）。更新前イメージチェック手段14は、収集情報テーブル23bの全ての情報を調べ、ブロック番号21は登録されていないため（ステップ401）、収集情報テーブル23bにブロック番号21を追加し（ステップ402）、ファイルアクセス手段12に戻る。

【0078】ファイルアクセス手段12は、ブロック番号21が収集されていないため、更新前イメージ収集手段15を動作させる（ステップ205、206）。更新50 前イメージ収集手段15は、ブロック番号21をデータ

ファイル34から読み込み、その内容を収集ファイル35bのブロック番号43bに書き出し（ステップ501、504、503）、ファイルアクセス手段12に戻る。

【0079】ファイルアクセス手段12は、ファイルI/O手段16を動作させ（ステップ207）、データファイル34を更新する。ファイルI/O手段16は更新であるため（ステップ601）、プログラム11bで指定された内容をデータファイル34のブロック番号21へ書き出し（ステップ605）、プログラム11bに戻る。

【0080】図示の状態の時、プログラム11aがブロック番号25を更新しようとした時、ファイルアクセス手段12は、排他制御手段13を動作させる（ステップ201、202）。排他制御手段13は、ロック要求であるため（ステップ301）、排他制御テーブル22を調べると、ブロック番号26はプログラム11aしかブロック番号25をロックしていないため、アクセス可能と判断する（ステップ302）。そして、更新であるため（ステップ303）、排他制御テーブル22のプログラム11a用のテーブルにブロック番号25を更新ロックとして追加するが、既にテーブルに存在するため、何もせず（ステップ305）にファイルアクセス手段12に戻る。

【0081】ファイルアクセス手段12は、次に更新前イメージチェック手段14を動作させる（ステップ203、204）。更新前イメージチェック手段14は、収集情報テーブル23aの全ての情報を調べ、ブロック番号25が登録されているため（ステップ401）、何もせずにファイルアクセス手段12に戻る。

【0082】ファイルアクセス手段12は、ブロック番号25が既に収集されているため（ステップ205）、ファイルI/O手段16を動作させる（ステップ207）。ファイルI/O手段16は更新であるため（ステップ601）、プログラム11aで指定された内容をデータファイル34のブロック番号25へ書き出し（ステップ606）、プログラム11aに戻る。

【0083】図示の状態の時、プログラム11bがブロック番号25を更新しようとした時、ファイルアクセス手段12は、排他制御手段13を動作させる（ステップ201、202）。排他制御手段13は、ロック要求であるため（ステップ301）、排他制御テーブル22を調べると、ブロック番号25はプログラム11a用のテーブルに存在し（ステップ302）、参照どうしでなく（ステップ306）、ロールバック中の表示はないため（ステップ307）、プログラム11aがブロック番号25をアンロックするまで待ち状態となる（ステップ308）。

【0084】同様に図示の状態の時、プログラム11aがチェックポイントを取った時、ファイルアクセス手段

12は、チェックポイント要求であるため（ステップ201）、チェックポイント手段17を動作させる（ステップ208）。チェックポイント手段17はまず排他制御手段13を動作させ（ステップ701）、アンロック要求を出す。排他制御手段13は、アンロック要求であるがロールバック開始前ではないため（ステップ312）、排他制御テーブル22のプログラム11aのテーブルにロックされていたブロック番号23、25、26の情報を削除する（ステップ313）ことによって、ロックしていた全てのブロックをアンロックして、チェックポイント手段17に戻る。チェックポイント手段17は、次に収集情報テーブル23aに登録されていたブロック番号26、26の情報を削除する（ステップ702）ことによって、収集情報テーブル23aの情報を初期化し、もしプログラム11aが異常終了してもロールバックによりデータファイル34の収集ファイルにあつたブロックの内容は書き戻されないようにした上でプログラム11aに戻る。

【0085】図示の状態の後、プログラム11zが動作させた更新取消手段18は、収集情報テーブル23zに登録してあるブロック（ブロック番号30、31、32）を一つずつ順に収集ファイル35zからデータファイル34へ書き戻す（ステップ801）。収集情報テーブルに存在した全てのブロックのデータを書き戻した後、排他制御手段13を動作させ（ステップ803）、アンロック要求を出す。

【0086】排他制御手段13は、アンロック要求であり（ステップ301）、ロールバック開始前でないため（ステップ312）、排他制御テーブル22のプログラム11z用のテーブルを削除し、ロールバック中の表示を消し（ステップ313）、プログラム11zは終了する。

【0087】以上好ましい実施の形態と実施例をあげて本発明を説明したが、本発明は必ずしも上記実施の形態及び実施例の内容に限定されない。

【0088】

【発明の効果】以上説明したように本発明の排他制御方式によれば、以下に述べるような効果が得られる。

【0089】第1に、あるプログラムがロールバック中のブロックに対する更新と他のプログラムによる同一ブロックへの同時アクセスが可能となり、スループット向上が実現される。その理由は、ロールバックの開始前にロールバックするブロックを全てアンロックし、他のプログラムからのそのブロックに対するアクセスは収集ファイルに対して行うためである。

【0090】第2に、従来のような更新プログラムを中止しなければならない状態に陥ることがなく、更新プログラムの中止による処理時間の増加を回避することができる。その理由は、二相ロックを用いた多版同時実行制御を用いず、古いブロックの内容を参照することがない

ためである。

【0091】第3に、従来のようなデータファイルのサイズの増大が発生しないことである。その理由は、ロールバックに用いる収集ファイルを利用して参照を行なうため、更新するたびに新たにファイルスペースを確保する必要がないためである。

【図面の簡単な説明】

【図1】 本発明の実施の形態によるファイルアクセス処理システムの構成を示すブロック図である。

【図2】 本発明の実施の形態によるファイルアクセス処理システムのファイルアクセス手段の処理を示すフローチャートである。

【図3】 本発明の実施の形態によるファイルアクセス処理システムの排他制御手段の処理を示すフローチャートである。

【図4】 本発明の実施の形態によるファイルアクセス処理システムの更新前イメージチェック手段の処理を示すフローチャートである。

【図5】 本発明の実施の形態によるファイルアクセス処理システムの更新前イメージ収集手段の処理を示すフローチャートである。

【図6】 本発明の実施の形態によるファイルアクセス処理システムのファイルI/O手段の処理を示すフローチャートである。

【図7】 本発明の実施の形態によるファイルアクセス処理システムのチェックポイント手段の処理を示すフローチャートである。

【図8】 本発明の実施の形態によるファイルアクセス処理システムの更新取消手段の処理を示すフローチャートである。

【図9】 本発明の実施の形態によるファイルアクセス

処理システムの排他制御テーブルの内容を例示した図である。

【図10】 本発明の実施の形態によるファイルアクセス処理システムの収集情報テーブルの内容を例示した図である。

【図11】 本発明の実施の形態によるファイルアクセス処理システムの収集ファイルの内容を例示した図である。

【図12】 本発明の実施の形態によるファイルアクセス処理システムのデータファイルの内容を例示した図である。

【符号の説明】

10 1 データ処理装置

20 2 主記憶装置

30 3 補助記憶装置

11 1 プログラム

11a~11z 2 プログラム

12 3 ファイルアクセス手段

13 4 排他制御手段

20 14 更新前イメージチェック手段

15 15 更新前イメージ収集手段

16 16 ファイル1/0手段

17 17 チェックポイント手段

18 18 更新取消手段

22 22 排他制御テーブル

23 23 収集情報テーブル

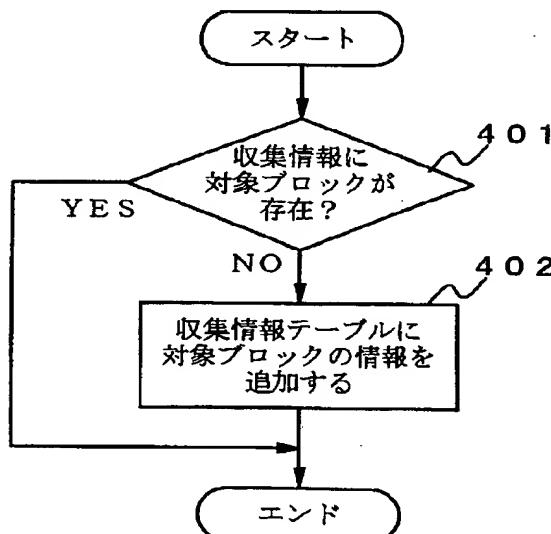
23a~23z 24 プログラム毎の収集情報テーブル

34 25 データファイル

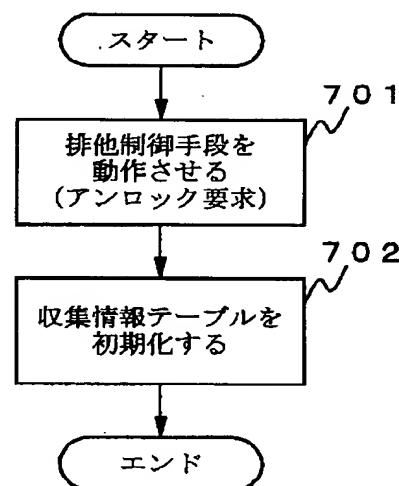
35 26 収集ファイル

35a~35z 27 プログラム毎の収集ファイル

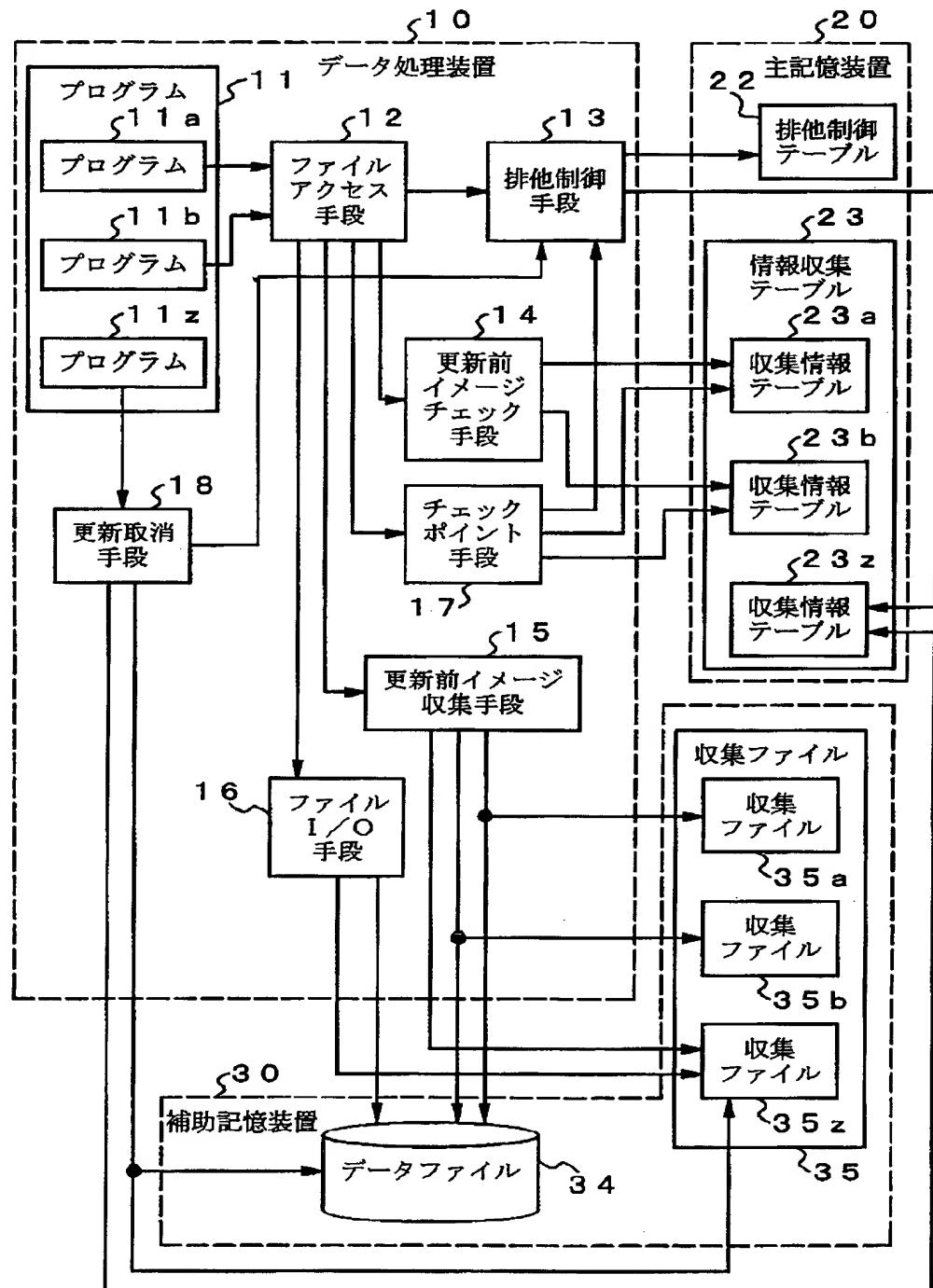
【図4】



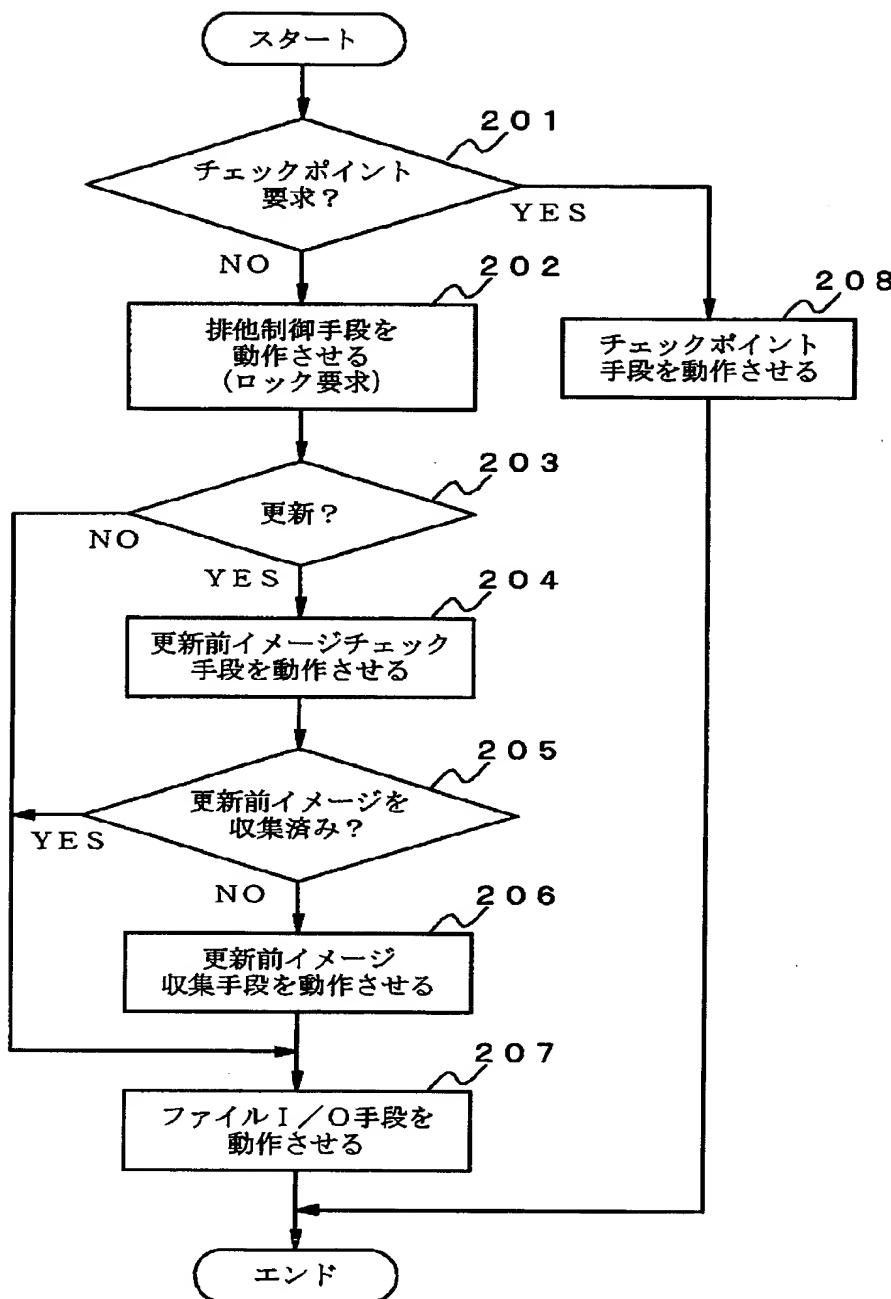
【図7】



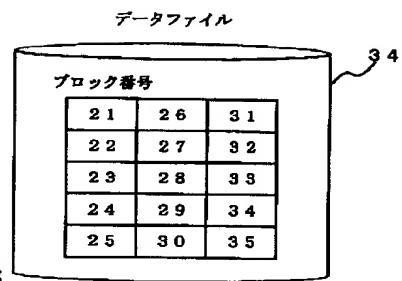
【図1】



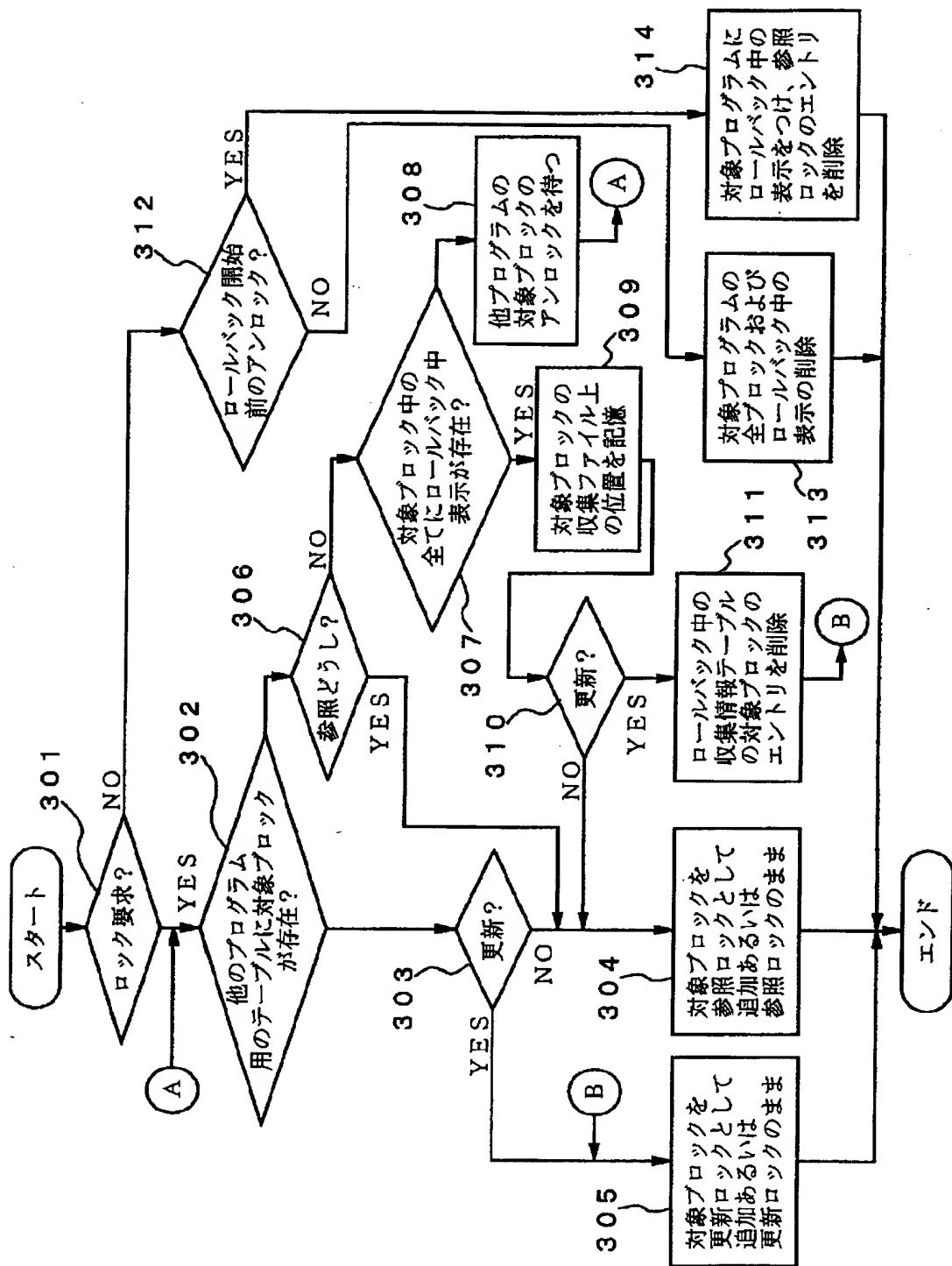
【図2】



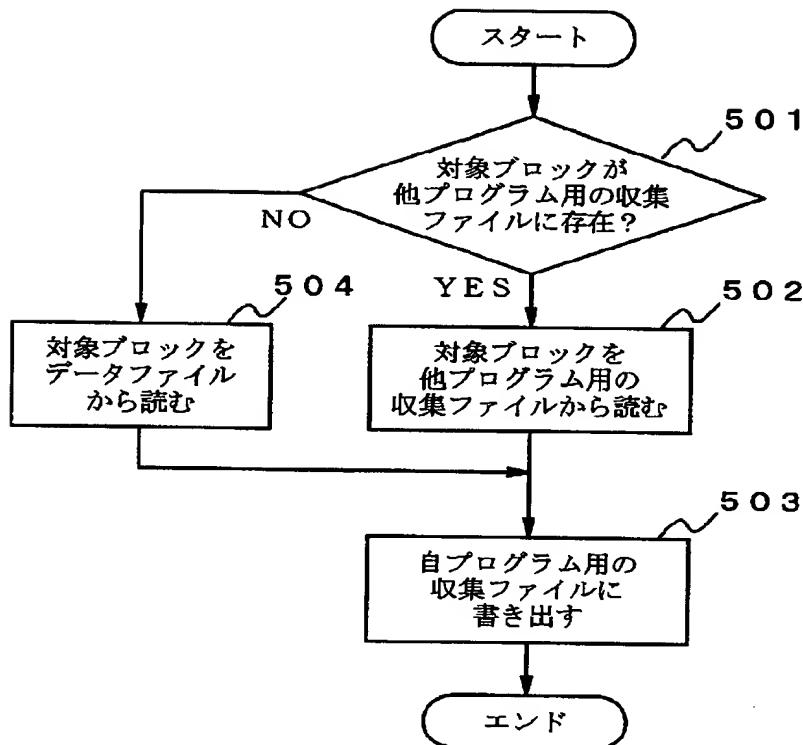
【図12】



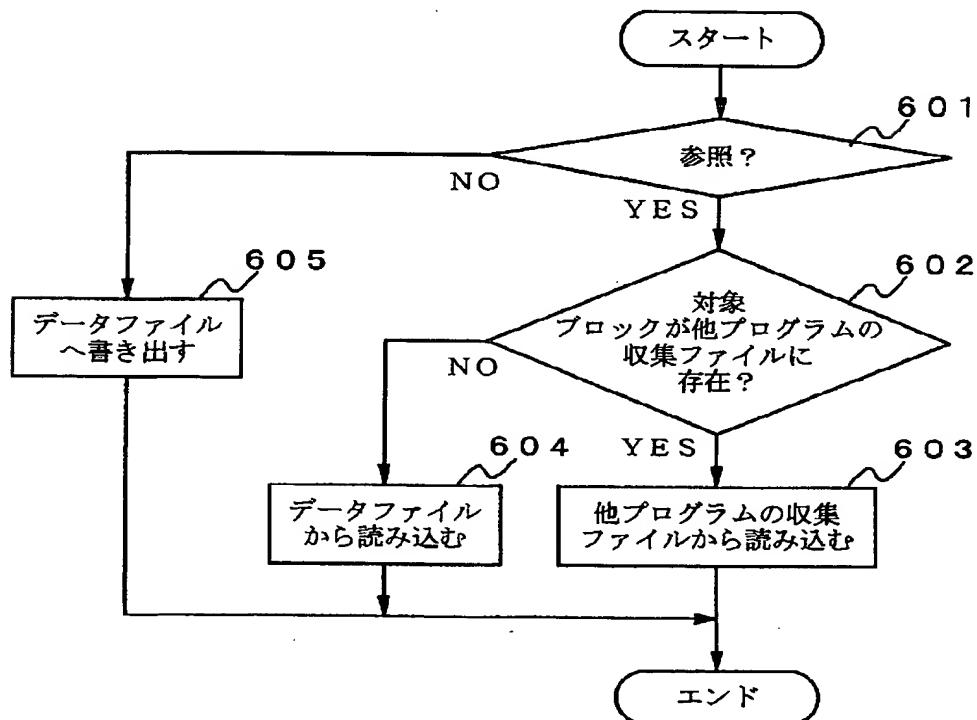
【図3】



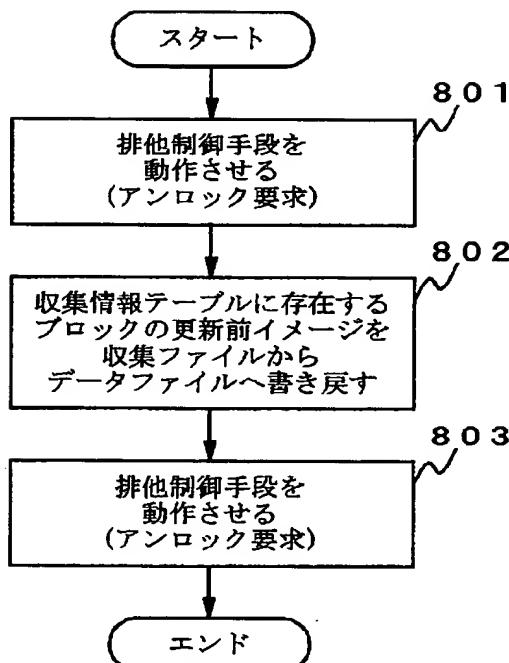
【図5】



【図6】



【図8】



【図10】

収集ファイルの位置 ブロック番号、 収集ファイルの ブロック番号	収集情報テーブル		
	23a (アマ'11a用)	23b (アマ'11b用)	23z (アマ'11z用)
25	41a	27	41b
26	42a	29	42b
空き	43a	空き	43b
空き	44a	空き	44b
	空き	49a	空き
		49b	空き
			49z

【図9】

排他制御テーブル		
アマ'11a用	アマ'11b用	アマ'11z用
なし	なし	なし
23 参照	27 更新	30 更新
25 更新	29 更新	31 更新
26 更新	空き	32 更新
空き	空き	空き
空き	空き	空き

【図11】

データファイルの ブロック番号	収集ファイル		
	35a (アマ'11a用)	35b (アマ'11b用)	35z (アマ'11z用)
ブロック番号 25の内容	ブロック番号 27の内容	ブロック番号 30の内容	ブロック番号 30の内容
ブロック番号 26の内容	ブロック番号 29の内容	ブロック番号 31の内容	ブロック番号 31の内容
空き	空き	空き	空き
空き	空き	空き	空き
空き	空き	空き	空き

**This Page is Inserted by IFW Indexing and Scanning
Operations and is not part of the Official Record**

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- BLACK BORDERS**
- IMAGE CUT OFF AT TOP, BOTTOM OR SIDES**
- FADED TEXT OR DRAWING**
- BLURRED OR ILLEGIBLE TEXT OR DRAWING**
- SKEWED/SLANTED IMAGES**
- COLOR OR BLACK AND WHITE PHOTOGRAPHS**
- GRAY SCALE DOCUMENTS**
- LINES OR MARKS ON ORIGINAL DOCUMENT**
- REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY**
- OTHER:** _____

IMAGES ARE BEST AVAILABLE COPY.

As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.